10/031039

C 07 C.

CHARLES B. GORDON
THOMAS P. SCHILLER
DAVID B. DEIOMA
JOSEPH J. CORSO
HOWARD G. SHIMOLA
JEFFREY J. SOPKO
JOHN P. MURTAUGH
JAMES M. MOORE
MICHAEL W. GARVEY
RICHARD A. SHARPE
RONALD M. KACHMARIK
PAUL A. SERBINOWSKI
BRIAN G. BEMBENICK
AARON A. FISHMAN

PEARNE & GORDON LLP

ATTORNEYS AT LAW
1801 EAST 9th STREET
SUITE 1200
CLEVELAND, OHIO 44114-3108
TEL: +1 (216) 579-1700 FAX: +1 (216) 579-6073
EMAIL: ip@pearnegordon.com

STEPHEN S. WENTSLER
ROBERT F. BODI
SUZANNE B. GAGNON
UNA L. LAURICIA
STEVEN J. SOLOMON
GREGORY D. FERNENGEL
BRYAN M. GALLO
BRAD C. SPENCER
OF COUNSEL
LOWELL L. HEINKE
THADDEUS A. ZALENSKI
PATENT AGENT
TOMOKO ISHIHARA

PATENT, TRADEMARK, COPYRIGHT AND RELATED INTELLECTUAL PROPERTY LAW

August 29, 2006

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Re: U.S. Patent No. 7,079,654
Issued: July 18, 2006
Inventor: Remery et al.
Our Docket No.: 34333

Sir:

A Certificate of Correction under 35 U.S.C. 254 is hereby requested to correct Patent Office printing errors in the above-identified patent. Enclosed herewith is a proposed Certificate of Correction (Form No. PTO-1050) and documentation in support of the proposed corrections for consideration.

It is requested that the Certificate of Correction be completed and mailed at an early date to the undersigned attorney of record. The proposed corrections are obvious ones and do not in any way change the sense of the application.

We understand that a check is not required since the errors were on the part of the Patent and Trademark Office in printing the patent.

Very truly yours,

Jeffrey J. Sopko, Reg. No. 27676

JJS:ljw

Enclosures

I hereby certify that this correspondence is being deposited with the United States Postal Service as first class mail in an envelope addressed to: Commissioner of Patents, P.O. Box 1450, Alexandria, VA 22313-1450 on the date indicated below.

Jeffrey J. Sopko
Name of Attorney for Applicant(s)

8/29/06
Date **Signature of Attorney**

SEP 06 2006

PTO/SB/44 (07-03)
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
(Also Form PTO-1050)

UNITED STATES PATENT AND TRADEMARK OFFICE
CERTIFICATE OF CORRECTION

PATENT NO. : 7,079,654 PAGE 1 OF 1
DATED : July 28, 2006
INVENTOR(S) : Remery et al.

It is certified that error appears in the above-identified patent and that said Letters Patent is hereby corrected as shown below:

In Column 5, line 53, please delete "(n-2)" and insert - -(n-2) th- -

In Column 5, line 54, please delete "(n-1)" and insert - -(n-1)th- -

In Column 5, line 55, please delete "(n-1)" and insert - -(n-1)th- -

In Column 6, line 10, please delete "$K_{1,3}$ ($K_{1,4}$ (. . . ($K_{1,n-1}$ ($m_{1,m}$), $m_{1,n-1}$) , . . . $m_{1,4}$) , $M_{1,3}$) and insert - -$K_{1,3}$ ($K_{1,4}$ (. . . ($K_{1,n-1}$ ($m_{1,n}$), $m_{1,n-1}$) , . . . $m_{1,4}$) , $M_{1,3}$)- -

In Column 6, line 18, please delete "$K_{1,i+1}$ ($K_{1,i+2}$(...($K_{1,n+1}$ ($K_{1,n}$($m_{1,n}$)),$m_{1,n-1}$),...$m_{1,i+2}$),$m_{1,i+1}$)" and insert - -$K_{1,i+1}$($K_{1,i+2}$(...($K_{1,n-1}$($K_{1,n}$($m_{1,n}$)),$m_{1,n-1}$),...$m_{1,i+2}$),$m_{1,i+1}$)- -

In Column 7, line 24, please delete "$K_{1,2}$(...$K_{1,i}$($K_{2,j+2}$(...$K_{1,n}$($m_{1,n}$),...$m_{1,j+1}$),$m_{1,i}$)...$m_{1,2}$)" and insert - -$K_{1,2}$(...$K_{1,i}$($K_{1,j+1}$(...$K_{1,n}$($m_{1,n}$),...$m_{1,j+1}$),$m_{1,i}$)...$m_{1,2}$)- -

In Column 8, line 16, please delete "$K_{i,i+1}$($m_{i,i+1}$),$K_{i,i+2}$($m_{i,i+2}$),...$K_{i,1}$($m_{i,1}$),.$m_{1,j+1}$" and insert - -$K_{i,i+1}$($m_{i,i+1}$),$K_{i,i+2}$($m_{i,i+2}$),...$K_{i,1}$($m_{i,1}$),$K_{1,j+1}$(...$K_{1,n}$($m_{1,n}$),...$m_{1,j+1}$- -

In Column 8, line 33, please delete "$K_{2,j+1}$(...$K_{1,n}$($m_{1,n}$),...$m_{1,j+1}$)" and insert - -$K_{1,j+1}$(...$K_{1,n}$($m_{1,n}$),...$m_{1,j+1}$)- -

In Column 9, lines 14, 15, 16 and 17 please insert the following:

- -MS
MS increments its
counter:
NT_{MS}=NT_{MS}+1- -

MAILING ADDRESS OF SENDER: Jeffrey J. Sopko
Pearne & Gordon LLP
1801 East 9th Street
Suite 1200
Cleveland, Ohio 44114-3108

PATENT NO. 7,079,654 B2

No. of additional copies
⇨ 0

"encapsulation" to describe the use of a cryptogram $X=K'(m)$ as a message.

- When cryptogram $K_{ij}(X,m)$ is intended to preserve the message's integrity but the channels that it must follow are controlled by entities that have an interest in X being transferred, we can obtain $K_{ij}(K(m'),m)=K(m')||K'_{ij}(m)$, for example, where there are no restrictions to which algorithm may be used to calculate the cryptograms.

We will describe four examples of preferred embodiments for this method:

Example 1: Total encapsulation at source and progressive decapsulation

The source builds a message $m_{1,n}$ combining all of the transaction data and calculates a first cryptogram $K_{1,n}(m_{1,n})$ of this first message using a first key system $K_{1,n}$ that it shares with the last n^{th} entity; the source then links a second message $m_{1,n-1}$ with the first cryptogram and calculates a second cryptogram $K_{1,n-1}(K_{1,n}(m_{1,n}),m_{1,n-1})$ of the whole using a second key system $K_{1,n-1}$ that it shares with the last but one $(n-1)^{th}$ entity, and so on; the first entity links an $(n-1)^{th}$ message $m_{1,2}$ with the $(n-2)^{th}$ cryptogram previously obtained and calculates an $(n-1)^{th}$ cryptogram of the whole using the $(n-1)^{th}$ key system $K_{1,2}$ that it shares with a second entity; the source then sends the last calculated cryptogram across the communication network to entity 2.

We can represent this first stage in the following diagram, where the arrow pointing towards the right symbolises information being transferred between entity 1 (left) and entity 2 (right):

Entity 1 Entity 2

$K_{1,2}(K_{1,3}(K_{1,4}(\ldots(K_{1,n-1}(K_{1,n}(m_{1,n}),m_{1,n-1}),\ldots m_{1,4}),m_{1,3}),m_{1,2})$

---------->

Entity 2, which receives the message from entity 1, partially decapsulates this message using key system $K_{1,2}$; entity 2 checks (and possibly stores) the cryptogram intended for it (in this case the signature of message $m_{1,2}$), then sends the rest of the message to entity 3. Using the same conventions, we therefore obtain the following diagram:

Entity 2 Entity 3

$K_{1,3}(K_{1,4}(\ldots(K_{1,n-1}(K_{1,n}(m_{1,n}),m_{1,n-1}),\ldots m_{1,4}),m_{1,3})$

---------->

This method is then repeated so that the message gradually reaches entity n. For the intermediate entities i and i+1, we obtain:

Entity i Entity i+1

$K_{1,i+1}(K_{1,i+2}(\ldots(K_{1,n-1}(K_{1,n}(m_{1,n})),m_{1,n-1}),\ldots m_{1,i+2}),m_{1,i+1})$

---------->

Lastly, the last but one entity (n-1) sends the last cryptogram $K_{1,n}(m_{1,n})$ to recipient (n) which uses key system $K_{1,n}$ to retrieve the message intended for it:

Entity n-1 Entity n

Entity i — Entity i+1

$$K_{i,i+1}(K_{i,i+2}(K_{i,i+3}(\ldots(K_{i,n-1}(K_{i,n}(m_{i,n})),m_{1,n-1}),\ldots m_{i,i+3}),m_{i,i+2}),m_{i,i+1})$$

---------->

and so on through the entities of the second group until the last but one entity, n-1, which sends the last cryptogram to recipient n.

Example 3: General scenario

Entity 1 shares a key system with some of the entities on the communication route, which for the purposes of simplicity in this presentation we will suppose to be 2, ..., i, j+1, ..., n. Entity 1 therefore partially encapsulates the data as shown in the following diagram:

Entity 1 — Entity 2

$$K_{1,2}(\ldots K_{1,i}(K_{1,j+1}(\ldots K_{1,n}(m_{1,n}),\ldots m_{1,j+1}),m_{1,i})\ldots m_{1,2})$$

---------->

Each intermediate entity uses the appropriate key system to decapsulate the message that it receives, until the message reaches entity i:

Entity i-1 — Entity i

$$K_{1,i}(K_{1,j+1}(\ldots K_{1,n}(m_{1,n}),\ldots m_{1,j+1}),m_{1,j})$$

---------->

Each actor (in this case, only "i") extracts the message sent to it, so obtaining the remainder of the message intended for an actor that is not adjoining it on the route, and then re-encapsulates it for the adjoining entity and any following entities.

no reason for the entities involved to falsify the messages.

Entity i — Entity i+1

$K_{i,i+1}(m_{i,i+1}), K_{i,i+2}(m_{i,i+2}), \ldots K_{i,1}(m_{i,1}), K_{1,j+1}(\ldots K_{1,n}(m_{1,n}), \ldots m_{1,j+1})$

---------->

Each intermediate entity receives and checks the message sent to it, using the key system, until the message reaches entity j.

Entity j-1 — Entity j

$K_{i,j}(m_{i,j}), K_{1,j+1}(\ldots K_{1,n}(m_{1,n}), \ldots m_{1,j+1})$

---------->

Entity j receives and checks the message sent to it. This message is then sent gradually from j+1 to n:

Entity j — Entity j+1

$K_{1,j+1}(\ldots K_{1,n}(m_{1,n}), \ldots m_{1,j+1})$

---------->

Entity n-1 — Entity n

$K_{1,n}(m_{i,n})$

---------->

Example 4: electronic wallet (PME)

In this example, the entities (or actors) are as follows:

- PME cards (A),
- service points (P) that are capable of receiving the cards,
- service point concentrators, together with their security module (MS),

A	P	MS
$K_{A,M}(NT_A, NT_{MS}, ID_{MS})$	$K_{A,M}(NT_A, NT_{MS}, ID_{MS})$	MS increments its counter: $NT_{MS}=NT_{MS}+1$
<----------	<----------	

Card A checks the data that it has received and resets the running total to zero (RunningTotal=0).

The service unit consumption cycle then begins. The following operations are then performed:

A P

order to debit amount m

<----------

RunningTotal: =RunningTotal+m

microtransaction calculated

$K_{A,P}(M, K_{A,M}(M, K_{A,E}(M')))$

---------->

where $M=(m, RunningTotal, NT_A, NT_{MS}, ID_{MS})$

and $M'=(RunningTotal, NT_A, NT_{MS}, ID_{MS})$

P checks the data that has been sent to it

RunningTotal: =RunningTotal+m

The process then returns to the beginning of the cycle if use of the service is not complete. At the end of the service session, the following final exchange takes place: